

- 45 -

**REPRESENTING TYPE INFORMATION IN A COMPILER AND
PROGRAMMING TOOLS FRAMEWORK**

5

ABSTRACT

A representation of types, type-checker, method and compiler are provided for checking consistency in various forms of an intermediate language. Specifically, the typed intermediate language is suitable for use in representing programs written in multiple (heterogeneous) source languages including typed and untyped languages, loosely and strongly typed languages, and languages with and without garbage collection. Additionally, the type checker architecture is extensible to handle new languages with different types and primitive operations. The representation of types can be implemented as a data structure that represents two or more sets of types with one or more sub-structures for representing classifications of types. Alternatively, the type system can be implemented as a base class at the top of a hierarchy and a plurality of classes hierarchically below the base class can be defined to represent a number of types such as pointer types, container types, and function types.